Synthetic Theater of WAR (STOW 97) Distributed Exercise Manager (DEM) Lessons Learned

Bernard Gajkowski, Greg Schow, STRICOM, AMSTI-ED, 12350 Research Parkway, Orlando, FL, 32826-3276

Jeff Swauger, Harvey Meier, Chris Gullette, TASC, Inc., 12443 Research Parkway, Suite 202, Orlando, FL 32826

John Pfalz, TASC, Inc., 4241 Woodcock Drive, Suite B-100, San Antonio, TX, 78228

Keywords:
HLA, DIS, STOW 97, MOM, SAF, RTI Monitoring, Run Time Analysis,
After Action Review

ABSTRACT: *The purpose of this paper is to report on the application and lessons learned of Distributed Exercise Management (DEM) through its participation in the Synthetic Theater Of War (STOW) 97 Advanced Concepts Technology Demonstration (ACTD). The DEM tool is unique in that through its scaleable hierarchical architecture, it was able to monitor the performance of ~ 500 federates (platforms) at different levels of detail in real-time. This ability provided the capability to identify problem areas more rapidly than has been possible to date. This in turn results in faster resolution of these areas of concern. DEM uses various means to collect the information it needs to monitor an exercise. Some of these are: platform systems calls, agents within the simulations, the Management Object Model (MOM) and Simple Network Monitoring Protocol (SNMP). The emphasis of DEM is the collection of information, not data. DEM restricted the collection of exercise events to those pieces of information required to understand the state of an exercise. Information collection thus was both more useful and required fewer resources for storage. DEM has taken a unique approach to exercise management by directly addressing the issues needed to provide an understanding of the exercise as it progresses. We will discuss the approach taken and lessons learned in STOW 97 by DEM. The STOW 97 ACTD is the first HLA and one of the largest virtual environment exercises that has been conducted.*

Background

The Distributed Exercise Manager (DEM) is one part of a larger U.S. Army Simulation Training and Instrumentation Command (STRICOM) sponsored program called Distributed Simulation Exercise Construction Toolset (DiSECT). DiSECT is part of the answer to a STRICOM initiative to reduce the cost of simulation 50% by 1999. DiSECT is split up into 3 main sections:

- Exercise Generation (ExGen)
- Distributed Exercise Manager (DEM)
- After Action Review (AAR)

The DEM portion of DiSECT is sponsored by both STRICOM and the Defense Advanced Research Projects Agency (DARPA) via the STOW 97 program. STOW was the first very large distributed exercise using the High Level Architecture (HLA), which is replacing Distributed Interactive Simulation (DIS). The current DEM design was heavily driven by STOW needs, and the outcome of a rigorous integration testing schedule. The culmination of DEM and the STOW 97 program came during the Advanced Concept Technical Demonstration (ACTD) in late October 1997. This paper discusses lessons learned by DEM during the STOW 97 integration testing and the ACTD.

DEM Objectives for STOW 97

DEM was conceptualized by STRICOM as a major portion of the continuous thread from pre simulation exercise activities (ExGen), through the simulation exercise (DEM), to analysis after the exercise is complete (AAR) within the DiSECT program. STRICOM entered into an agreement with DARPA in April 1996 to leverage on each other's needs with respect to DEM, and develop certain functionality for the ongoing STOW 97 program. This resulted in defining the following 4 main objectives for DEM:

- Provide Exercise Control for a large HLA based simulation.
- Monitor the health of the Run Time Infrastructure (RTI).
- Monitor the health of the simulation work stations
- Monitor the health of specific network parameters.

The Exercise Control component provided all of the functionality required to initialize and execute a federation within the STOW 97 HLA simulation environment. The functionality of the Exercise Controller provides the ability to Create, Destroy, Start, Stop, Pause, Resume, Save, and Restore a Federation. In the future, the ability to execute a magic move or a re-supply may be implemented to support adjudication. As the RTI matures, run-time exercise control will include the ability to change the loading on individual systems. The RTI provides methods which correspond to each of the aforementioned federation management functions. A graphical user interface has been designed to allow a simulation manager to execute these RTI methods. Within the HLA environment, it is the responsibility of the federates to honor requests from the RTI relating to these federation management functions. Therefore, correct operation of the DEM with regard to federation management requires each participating federate to implement these federation functions.

To ensure the proper execution of a federation within the HLA environment, an RTI Monitor is required to ensure correct operation of the constituent federates and the RTI itself. The following features were monitored using the Management Object Model (MOM) provided via the STOW 97 RTI:

1. List of federates (simulations) which have joined the federation.
2. Name of the federation the federate has joined.
3. Host machine name the federate is executing on.
4. Present state of the federation (paused or running).
5. Total number of packets received by the RTI.
6. Number of packets received by the RTI that are associated with a *reflectAttribute-Values()* call.
7. Percentage of traffic reduction associated with packet bundling within the RTI.
8. Average number of bytes in an RTI bundled packet.
9. Number of RTI multicast groups the federate is publishing information to.
10. Number of RTI multicast groups the federate is subscribing to.
11. Number of packets transmitted and received by the "Reliable" RTI transport mechanism.
12. Number of packets transmitted and received by the "Best Effort" RTI transport mechanism.
13. Number of packets transmitted and received by the "Minimum Rate" RTI transport mechanism.
14. Number of packets transmitted and received by the "State Consistent" RTI transport mechanism.
15. Number of negative acknowledgments received by a federate while using the "State Consistent" RTI transport mechanism.
16. Number of local objects known to the federate (sorted by class).
17. Number of remote objects known to the federate (sorted by class).

All items above were monitored and stored for use during run-time analysis and technical after action review.

DEM also had the responsibility for Network and CPU monitoring. This was accomplished by using DEM work stations at each STOW 97 site, or LAN to poll simulation hosts, and each other for information. The network traffic generated by the DEM polling was negligible. There were 3 methods DEM used to collect the data required to monitor CPU load, memory utilization, network traffic, and latency: SAF heartbeats, *rstatd*, and *ping*. The only information DEM monitored which it did not poll for was SAF frame rate. This was supplied from each simulation host on the LAN running a SAF application via a heartbeat. At startup of the SAF application, there is a command line option for the SAF to send information about it's frame rate to a pre-defined DEM workstation. This heartbeat is the frame rate at which the SAF is operating. SAFs are designed to run at 14-15 Hz when idle, but can run as low as 4 Hz without any degradation of fidelity. This heartbeat rate is variable, with a 15 second interval used for the STOW ACTD.

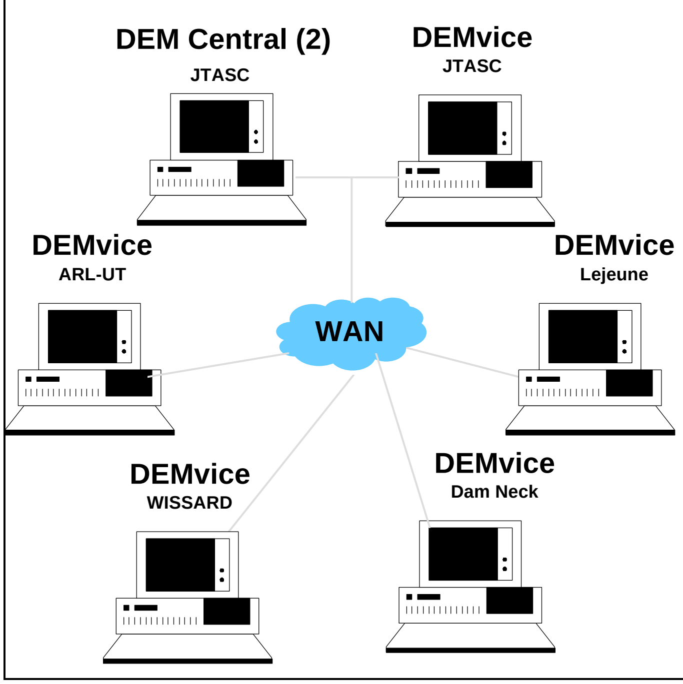
DEM used the UNIX *rstatd rpc* command to obtain CPU load monitoring data from simulation hosts. This also supplied DEM with work station memory and network traffic information. In all, *rstatd* supplied the following information to DEM:

1. Work Station packets in
2. Work Station packets out
3. Work Station errors in
4. Work Station errors out
5. Work Station network interface card (NIC) collisions
6. Work Station memory swap in
7. Work Station memory swap out
8. Work Station memory page in
9. Work Station memory page out
10. Work Station CPU utilization

All UNIX based machines on the LAN were polled by DEM (including the DEMvice itself) for this information at a rate of once every 15 seconds, and all this information was logged into a database. For UNIX machines running SAF applications, frame rate was also logged. DEM also monitored LAN-to-LAN latency and inter-connectivity.

As previously mentioned DEM provided four main functions. The DEM architecture design actually consists of a DEM Central and one or more DEMvices. DEM Central contains all exercise control functionality, does RTI monitoring, and supplies a central problem reporting repository for all of the data monitored by the DEMvices. DEMvices monitor MOM data, CPU load, memory utilization, and network traffic metrics for work stations connected on their LAN, while reporting problems to DEM central. The DEMvices and DEM Central also monitor latency and interconnectivity between LANs (see figure 1). The number of DEMvice computers required depends on how many LANs and how many simulation hosts per LAN exist in the simulation exercise. Normally, one DEMvice per LAN is recommended but more than one DEMvice can be used depending on the number of workstations on the LAN. DEM Central is located at the same location as the Exercise Coordinator. The workstation that hosts DEM Central can also host DEMvice depending on the load from the LAN/WAN traffic.

DEM STOW 97 ACTD Configuration



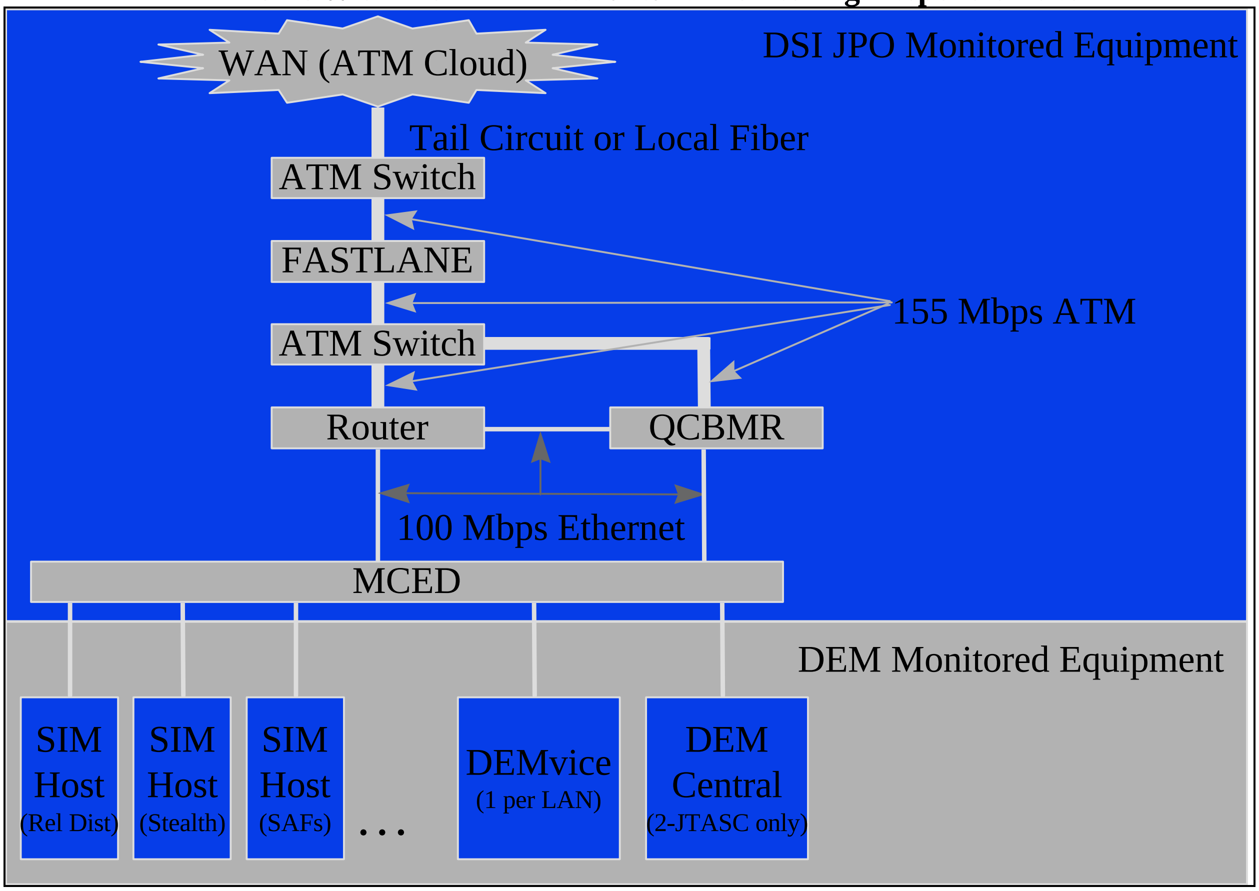
DEM Central:

- **Located at JTASC**
- **Monitor all RTI MOM Channels**
- **Provide HLA Exercise Control**
- **Process alarms from DEMvices**
- **Log exercise statistics**
- **LAN-to-LAN connectivity**
- **Remote data base query capability**
- **Receive MCED SNMP traps**

DEMvices:

- **Located at each simulation LAN**
- **Network load monitoring:**
  Packets in/out, Errors in/out, Collisions
- **Workstation monitoring:**
  CPU utilization, SAF frame rate
- **LAN-to-LAN Latency**
- **Monitor local RTI MOM Channel**
- **Alarms for out-of-tolerance conditions**
- **Log local LAN statistics**
- **Forward alarms to DEM Central**
- **Service DEM Central data requests**

DEM STOW 97 ACTD Infrastructure Monitoring Responsibilities



DEMvices were responsible for reporting and recording problems found during monitoring. When a problem was detected, an alarm (both visual and audible) was invoked locally (at the DEMvice), and an alarm message sent to DEM Central (using TCP/IP to guarantee delivery). If and when the problem corrected itself, the local alarm was cleared, and an alarm clear message sent to DEM Central. All alarm and alarm clear messages were written into alarm history files locally and at DEM Central. These files could be viewed by the DEMvice or DEM Central operator at any time for each monitored simulation host.

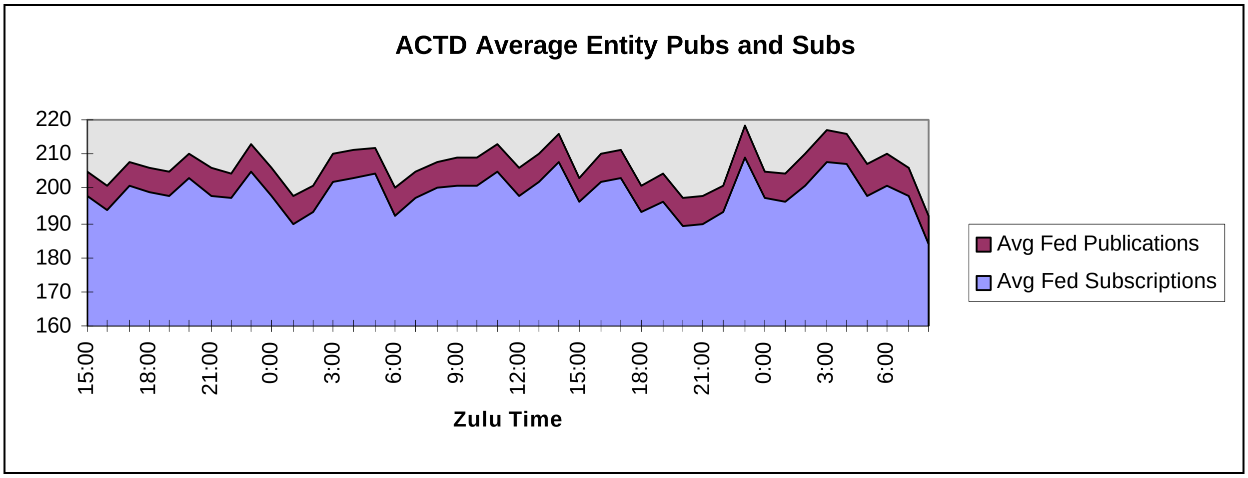
Accomplishments

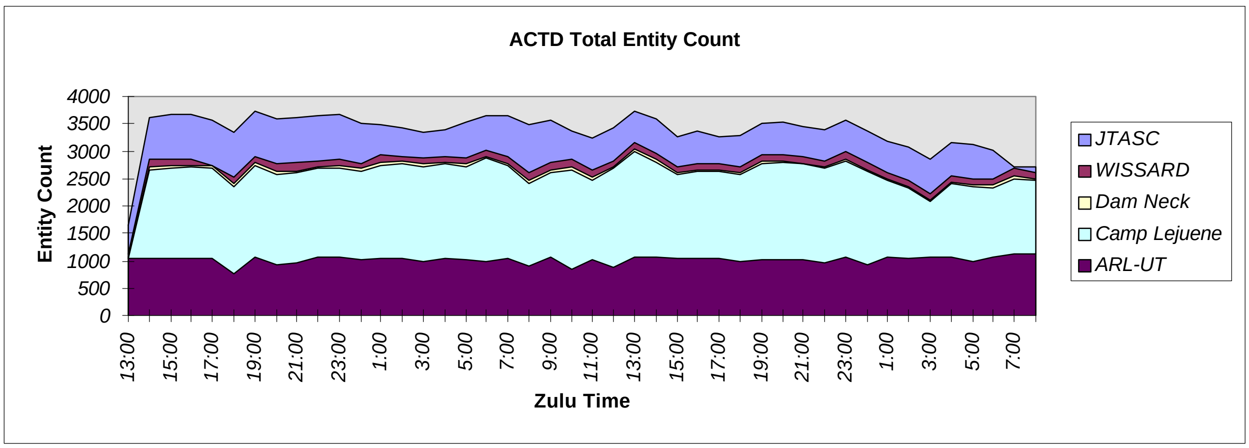
DEM was able to accomplish all of the objectives defined for the STOW 97 program. DEM completed it's objectives ahead of schedule, allowing further enhancements to be made before the ACTD. The extra enhancements made were primarily GUI enhancements and including RTI monitor capabilities in the DEMvices.

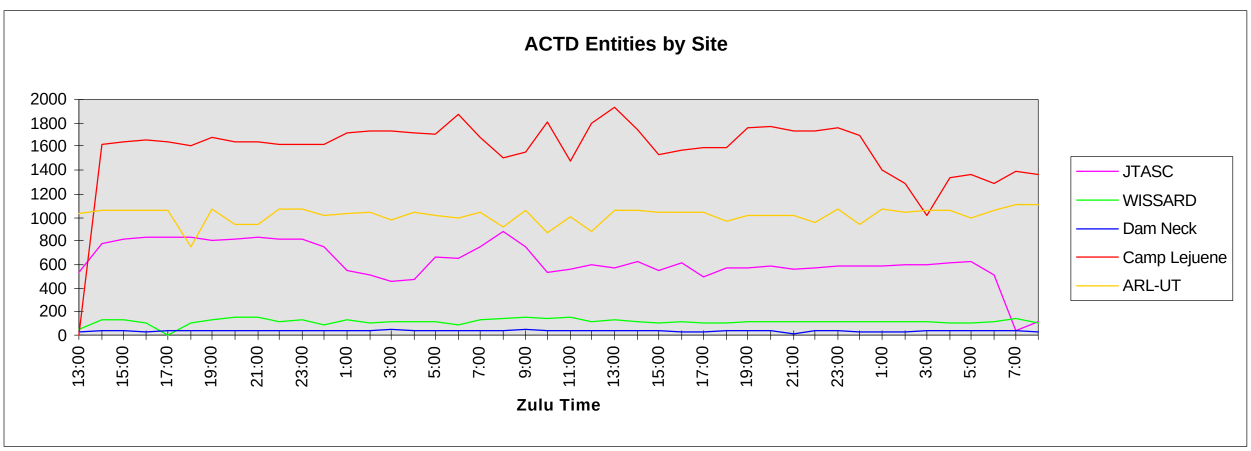
Capabilities Used During ACTD

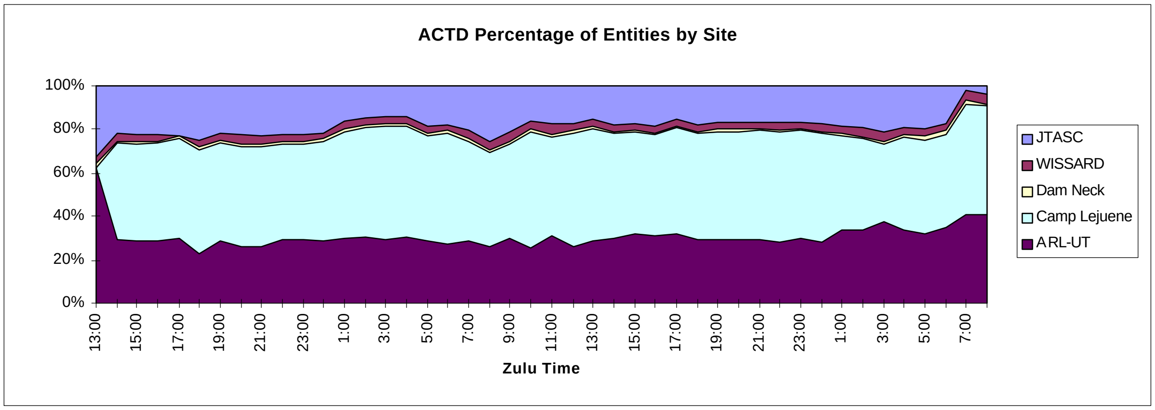
Not all of the DEM capabilities were taken advantage of during the ACTD. None of DEM's exercise control functionality was used during the ACTD. This is not necessarily bad, however because this functionality would be most likely used when problems occur during a simulation exercise. The STOW ACTD had no major show stopping problems which required the use of the DEM exercise control functionality.

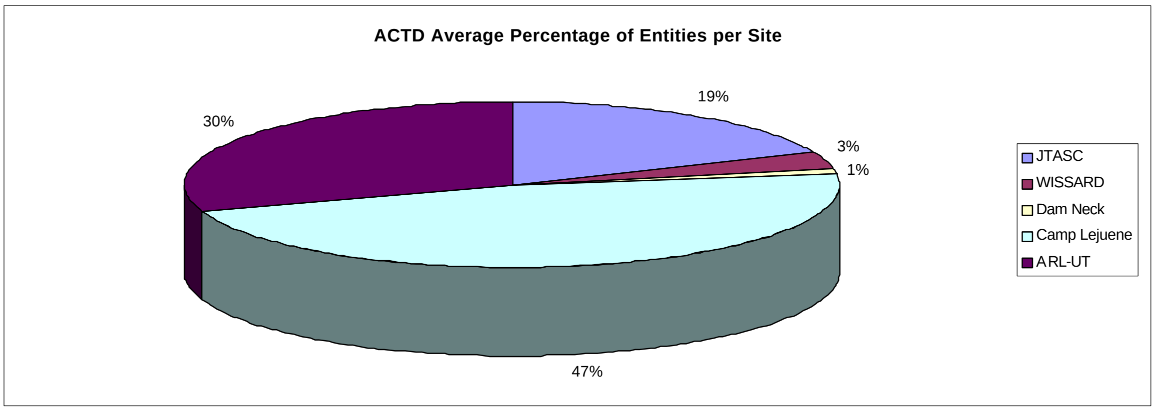
Most of the other DEM functionality was used in one way or another during the exercise. Frame rate monitoring was used consistently to monitor the health of the SAFs. Host network traffic was used to help debug performance problems. Latency was a good indication if a site was having problems. Certain RTI MOM data was used more than other MOM data. Entity counts and number of federates reporting were used more often than others. The following figures show data accumulated by DEM during the ACTD. The data collected is a snapshot of the situation on the hour; average hour counts were not taken. The time represented is in Zulu time.

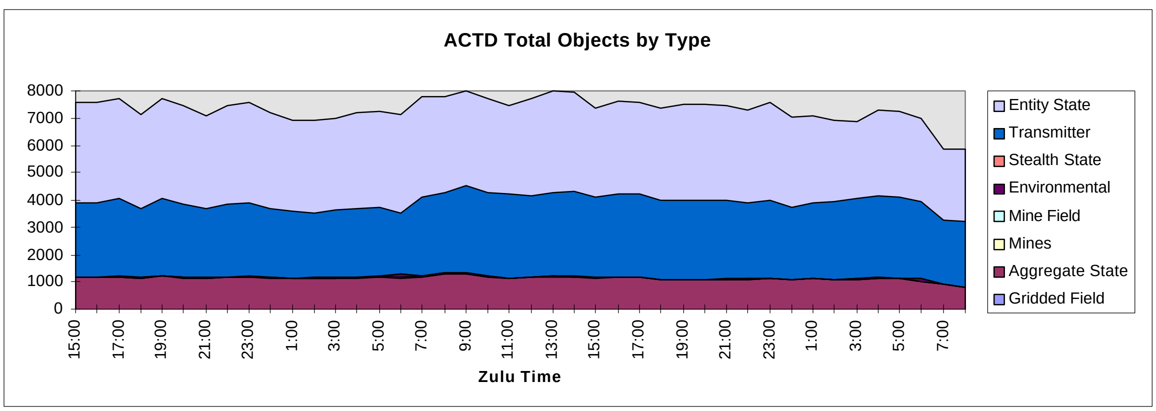

ACTD Average Entity Pubs and Subs
220
210
200
190
180
170
160
15:00
18:00
21:00
0:00
3:00
6:00
9:00
12:00
15:00
18:00
21:00
0:00
3:00
6:00
Zulu Time
Avg Fed Publications
Avg Fed Subscriptions

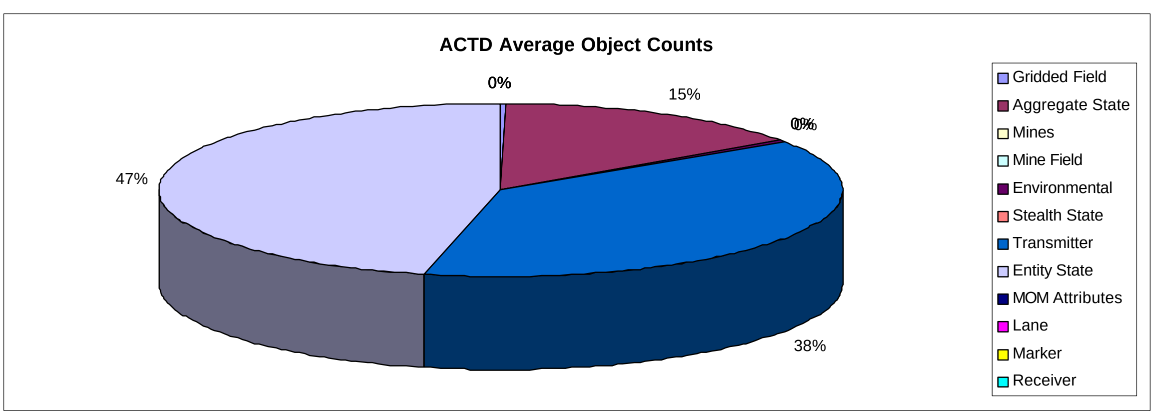

ACTD Total Entity Count
Entity Count
4000
3500
3000
2500
2000
1500
1000
500
0
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
9:00
11:00
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
Zulu Time
JTASC
WISSARD
Dam Neck
Camp Lejuene
ARL-UT

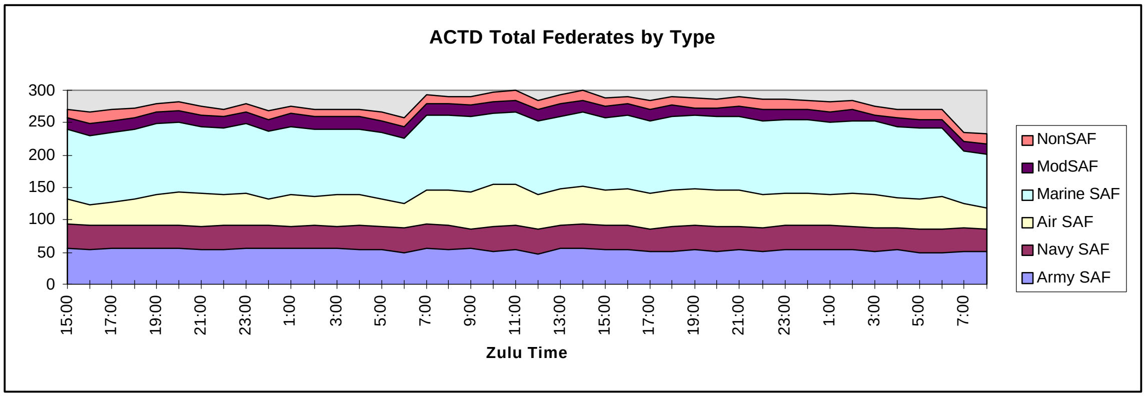

ACTD Entities by Site
2000
1800
1600
1400
1200
1000
800
600
400
200
0
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
9:00
11:00
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
Zulu Time
JTASC
WISSARD
Dam Neck
Camp Lejuene
ARL-UT

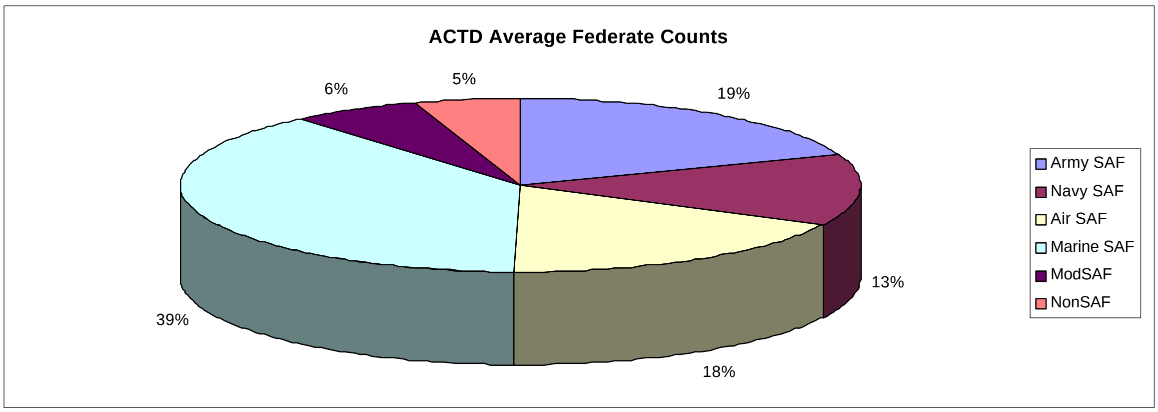

ACTD Percentage of Entities by Site
100%
80%
60%
40%
20%
0%
13:00 15:00 17:00 19:00 21:00 23:00 1:00 3:00 5:00 7:00 9:00 11:00 13:00 15:00 17:00 19:00 21:00 23:00 1:00 3:00 5:00 7:00
Zulu Time
JTASC
WISSARD
Dam Neck
Camp Lejuene
ARL-UT

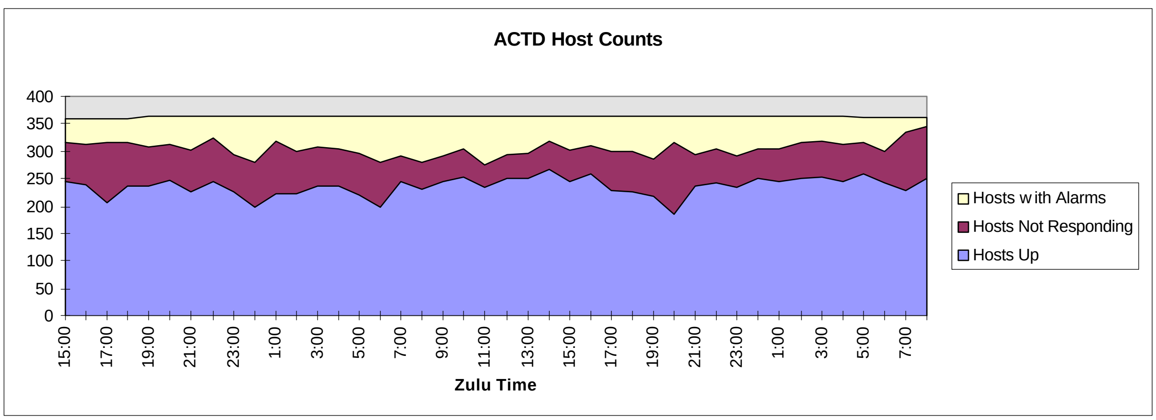

ACTD Average Percentage of Entities per Site
19%
3%
1%
30%
47%
JTASC
WISSARD
Dam Neck
Camp Lejuene
ARL-UT

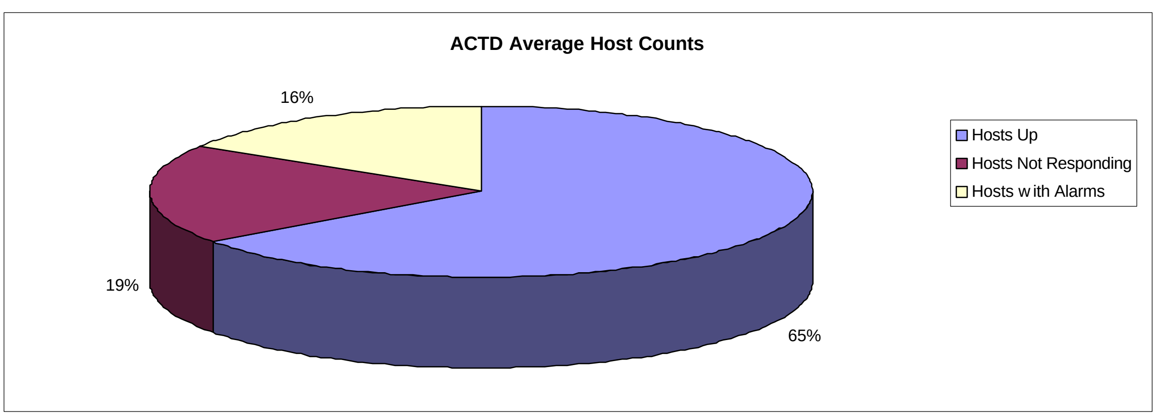

ACTD Total Objects by Type
8000
7000
6000
5000
4000
3000
2000
1000
0
15:00 17:00 19:00 21:00 23:00 1:00 3:00 5:00 7:00 9:00 11:00 13:00 15:00 17:00 19:00 21:00 23:00 1:00 3:00 5:00 7:00
Zulu Time
Entity State
Transmitter
Stealth State
Environmental
Mine Field
Mines
Aggregate State
Gridded Field

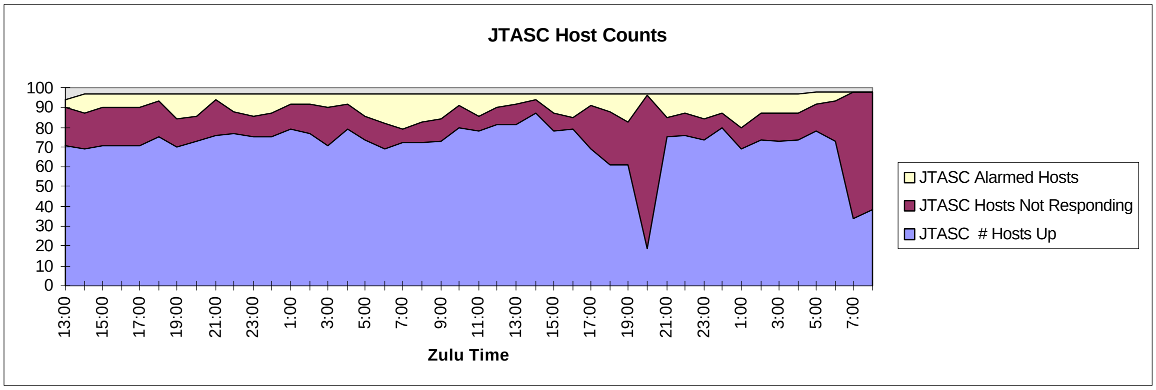

ACTD Average Object Counts
0%
15%
47%
38%
Gridded Field
Aggregate State
Mines
Mine Field
Environmental
Stealth State
Transmitter
Entity State
MOM Attributes
Lane
Marker
Receiver

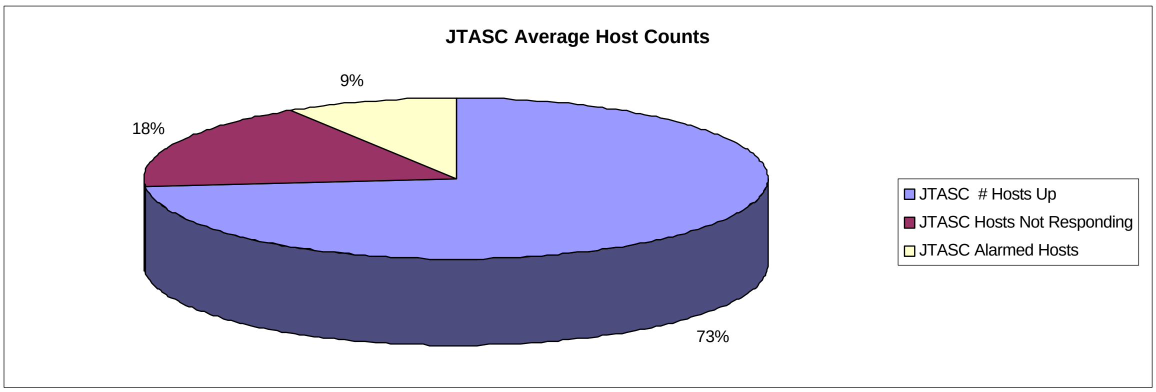

ACTD Total Federates by Type
300
250
200
150
100
50
0
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
9:00
11:00
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
Zulu Time
NonSAF
ModSAF
Marine SAF
Air SAF
Navy SAF
Army SAF

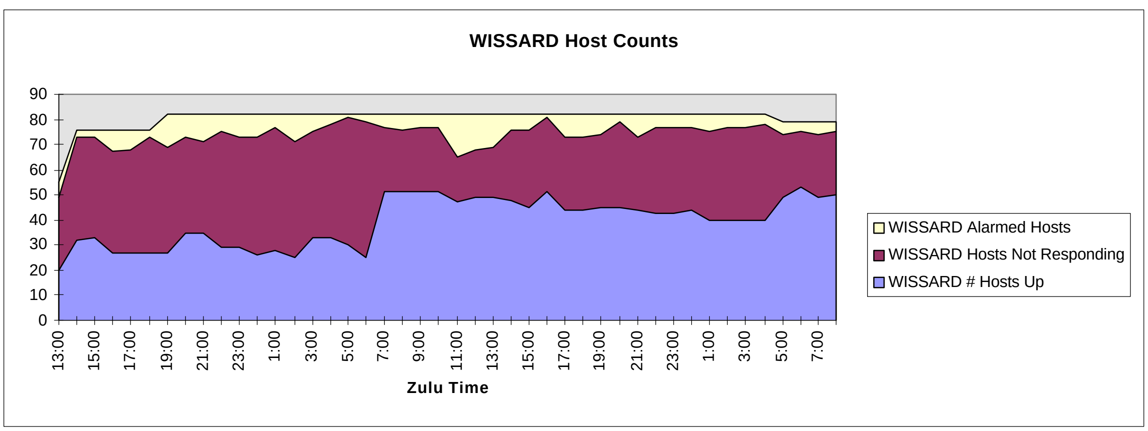

ACTD Average Federate Counts
6%
5%
19%
13%
39%
18%
Army SAF
Navy SAF
Air SAF
Marine SAF
ModSAF
NonSAF

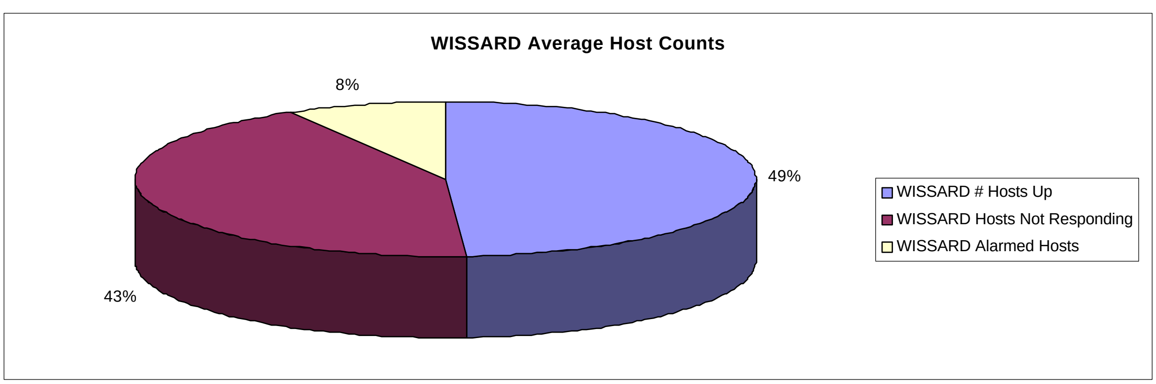

ACTD Host Counts
400
350
300
250
200
150
100
50
0
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
9:00
11:00
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
Zulu Time
Hosts with Alarms
Hosts Not Responding
Hosts Up

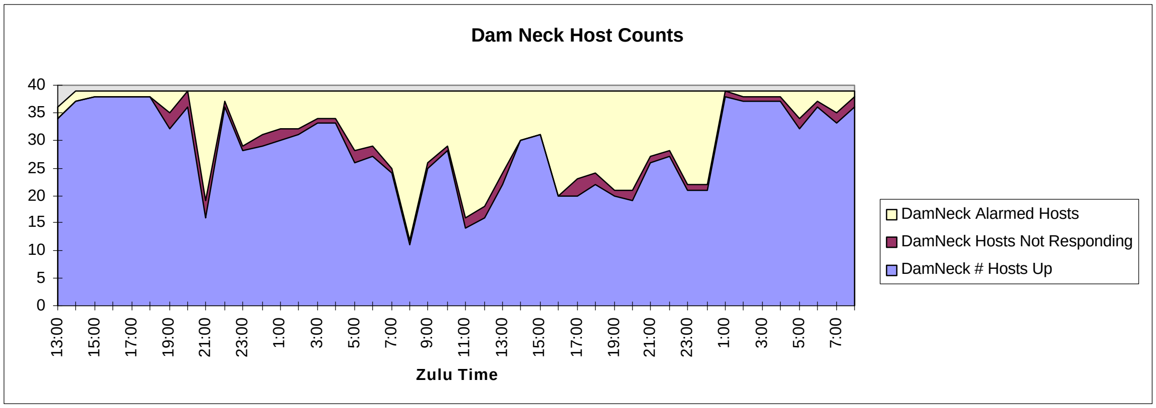

ACTD Average Host Counts
16%
19%
65%
Hosts Up
Hosts Not Responding
Hosts with Alarms

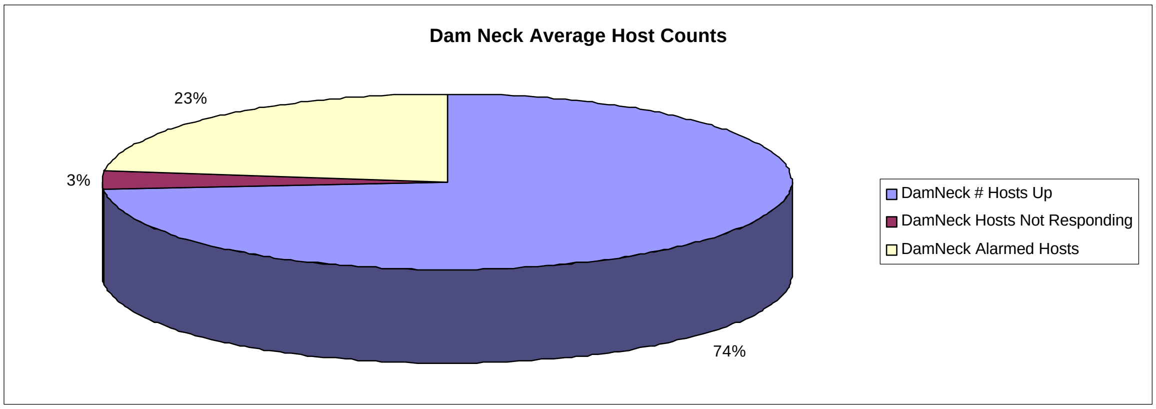

JTASC Host Counts
100
90
80
70
60
50
40
30
20
10
0
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
9:00
11:00
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
Zulu Time
JTASC Alarmed Hosts
JTASC Hosts Not Responding
JTASC # Hosts Up

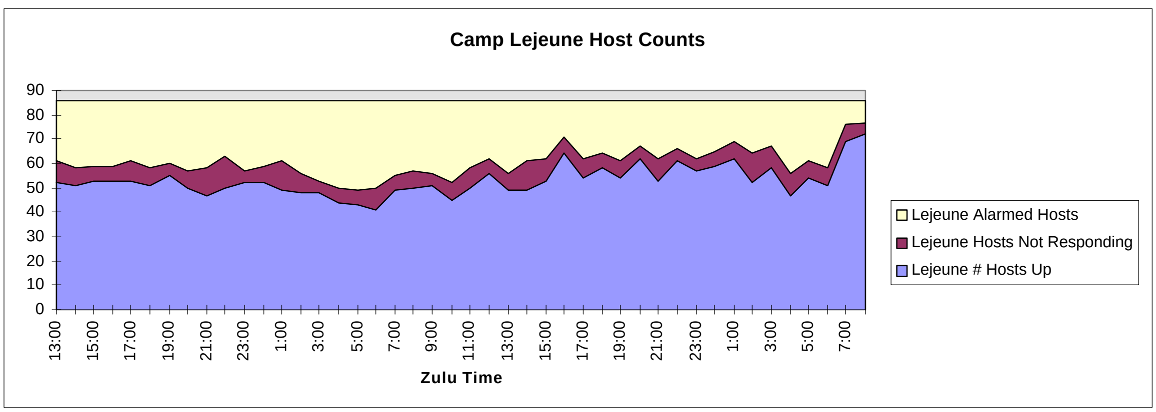

JTASC Average Host Counts
9%
18%
73%
JTASC # Hosts Up
JTASC Hosts Not Responding
JTASC Alarmed Hosts

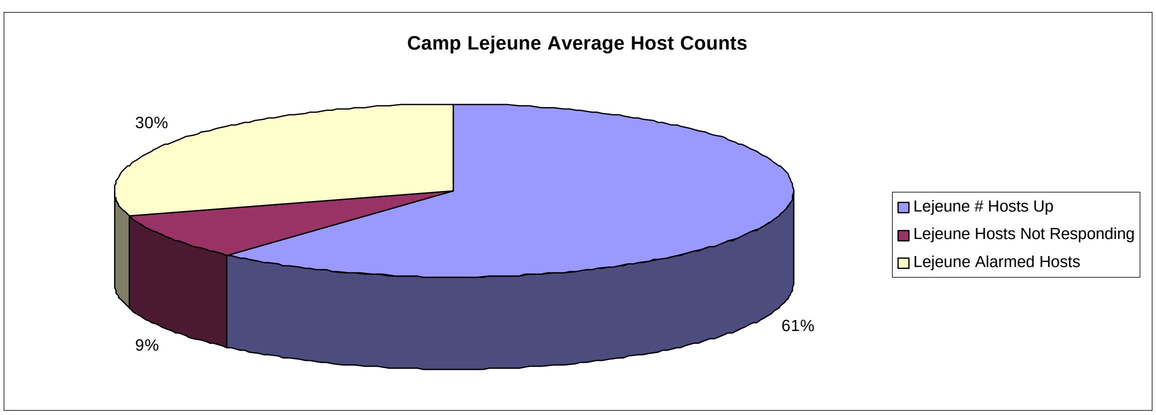

WISSARD Host Counts
90
80
70
60
50
40
30
20
10
0
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
9:00
11:00
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
Zulu Time
WISSARD Alarmed Hosts
WISSARD Hosts Not Responding
WISSARD # Hosts Up

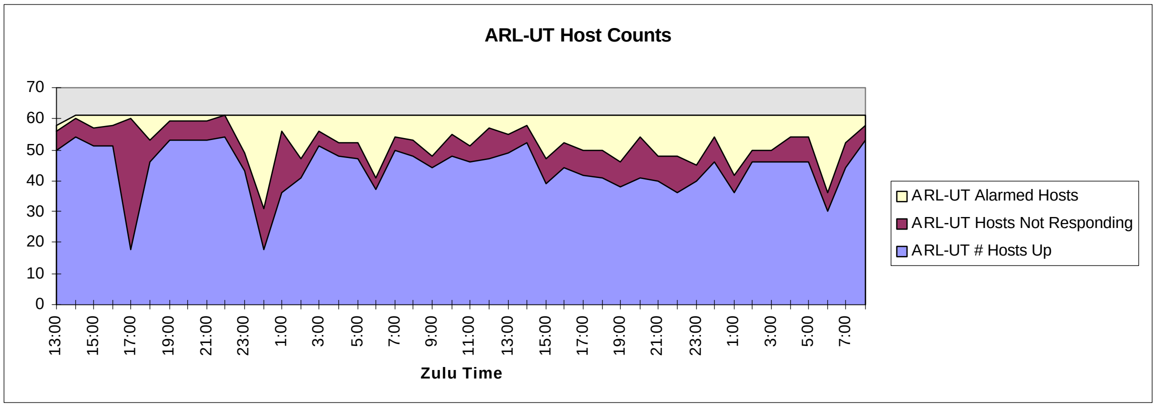

WISSARD Average Host Counts
8%
49%
43%
WISSARD # Hosts Up
WISSARD Hosts Not Responding
WISSARD Alarmed Hosts

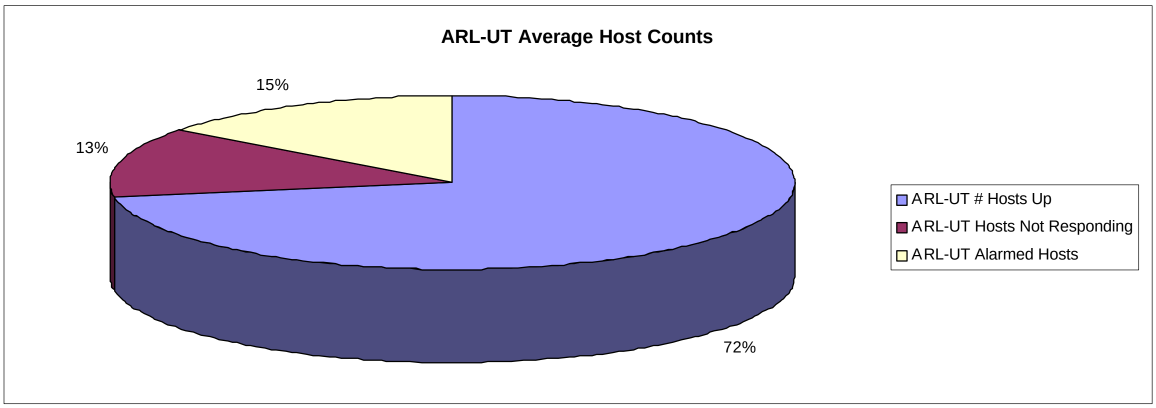

Dam Neck Host Counts
0
5
10
15
20
25
30
35
40
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
9:00
11:00
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
Zulu Time
DamNeck Alarmed Hosts
DamNeck Hosts Not Responding
DamNeck # Hosts Up

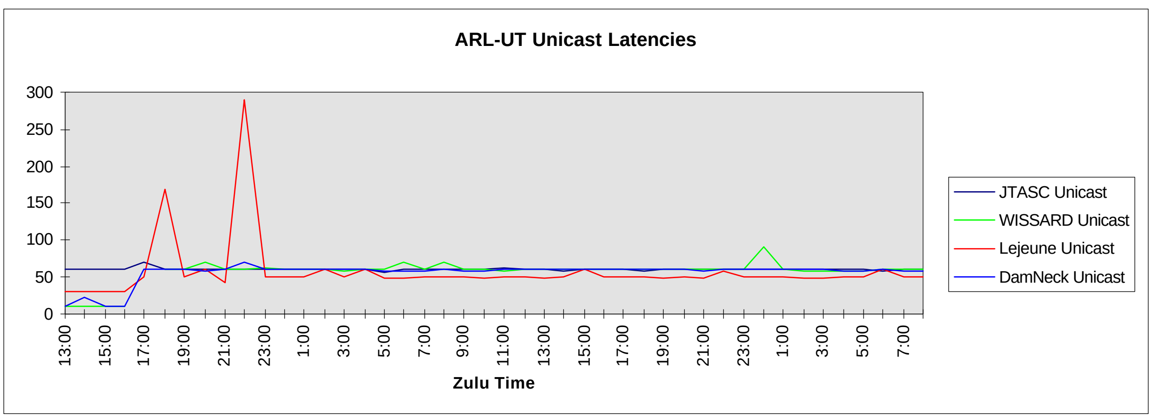

Dam Neck Average Host Counts
23%
3%
74%
DamNeck # Hosts Up
DamNeck Hosts Not Responding
DamNeck Alarmed Hosts

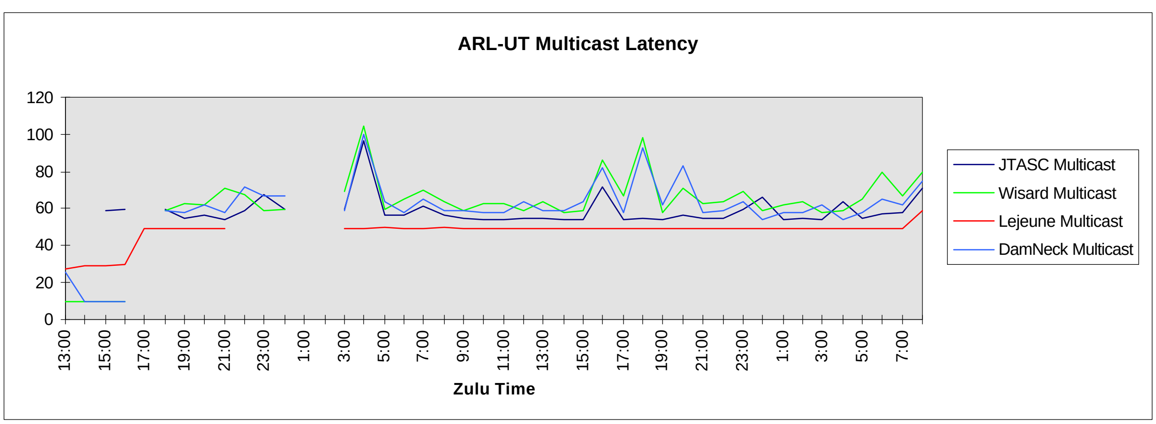

Camp Lejeune Host Counts
0
10
20
30
40
50
60
70
80
90
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
9:00
11:00
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
Zulu Time
Lejeune Alarmed Hosts
Lejeune Hosts Not Responding
Lejeune # Hosts Up


Camp Lejeune Average Host Counts
30%
9%
61%
Lejeune # Hosts Up
Lejeune Hosts Not Responding
Lejeune Alarmed Hosts


ARL-UT Host Counts
70
60
50
40
30
20
10
0
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
9:00
11:00
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
Zulu Time
ARL-UT Alarmed Hosts
ARL-UT Hosts Not Responding
ARL-UT # Hosts Up


ARL-UT Average Host Counts
15%
13%
72%
ARL-UT # Hosts Up
ARL-UT Hosts Not Responding
ARL-UT Alarmed Hosts


ARL-UT Unicast Latencies
300
250
200
150
100
50
0
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
9:00
11:00
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
Zulu Time
JTASC Unicast
WISSARD Unicast
Lejeune Unicast
DamNeck Unicast


ARL-UT Multicast Latency
120
100
80
60
40
20
0
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
9:00
11:00
13:00
15:00
17:00
19:00
21:00
23:00
1:00
3:00
5:00
7:00
Zulu Time
JTASC Multicast
Wisard Multicast
Lejeune Multicast
DamNeck Multicast

As shown in the figures above, there is a large variety of real-time information which DEM can display at any given time during an exercise. Some key observations made by DEM during the ACTD follow:

- Maximum Entities were just over 3700 - Lejeune (47%), ARL-UT (30%), JTASC (19%), WISSARD (3%), Dam Neck (1%)
- Federates Subscribed to an Average of 200 Multicast Groups and Published to an Average of 8 Multicast Groups
- Maximum Objects were just under 8000 - Entity State (47%), Transmitter (38%), Aggregate State (15%)
- Maximum of 300 Federates - Marine SAF (39%), Army SAF (19%), Air SAF 18%), Navy SAF (13%), ModSAF (6%), Non-SAF (5%)
- LAN to LAN Latencies (Unicast & Multicast) Averaged about 60ms from ARL-UT to all other sites
- LAN to LAN Latencies within the Norfolk area Averaged 10ms
- Average Hosts Monitored Information Over Entire ACTD:
- JTASC Hosts Up (73%), Alarmed (9%), Not Responding (18%)
- WISSARD Hosts Up (49%), Alarmed (8%), Not Responding (43%)
- Dam Neck Hosts Up (74%), Alarmed (23%), Not Responding (3%)
- Lejeune Hosts Up (61%), Alarmed (30%), Not Responding (9%)
- ARL-UT Hosts Up (72%), Alarmed (15%), Not Responding (13%)
- Overall Hosts Up (65%), Alarmed (16%), Not Responding (19%)

Problems / Lessons Learned

There were a few lessons learned for DEM as a direct result of being involved in the STOW program. These problems were not of great significance, however if corrected, could make DEM even more useful for a large simulation exercise. The following paragraphs describe the 3 main areas for improvement for DEM uncovered during STOW FST testing and the ACTD.

DEM was designed from the beginning to read a configuration file to determine which work stations it should monitor. Any changes in the simulation hardware infrastructure would require an update to the DEM configuration file. This was not considered a problem at first, but during FST testing and the ACTD there were many unexpected hardware configuration changes made. This made it difficult for DEM to keep an accurate count of host that were up, alarmed, or not responding.

Monitoring the large numbers of work stations during FST testing and the ACTD ended up with information overload for the DEM operators. The graphical representations displaying work station status were adequate, but did not make it very easy for the operator to quickly identify varying degrees of problems. Also, some data available from DEM was less likely to be used than other data.

DEM had a real time data base query capability which was tried during the ACTD. The system worked adequately at the beginning of the exercise when there was not very much data stored in the data base yet, however after 12 - 24 hours this system took far too long to reply to a data request. An operator could wait up to 15 minutes for an answer to his request. Although this will work fine in an AAR environment (this is actually what the system was designed for) it seemed to be unacceptable as a real time query tool.

Recommendations

The 3 main problems discussed in the previous section (Configuration file, GUI, and Real time data analysis) have been identified as DEM areas with room for improvement. The following paragraphs discuss recommendations for each.

It is recommended that DEM change it's host monitoring configuration setup from reading a static config file at startup to some sort of automatic configuration. The problem with a static config file is that if there are any changes which occur after DEM has started, the config file can not reflect these changes unless manually updated. During the STOW FST testing and the ACTD there was much more work station reconfiguring than was expected. For this reason, it is desirable to allow for a new work station to start up in the middle of a simulation exercise and have DEM automatically start to monitor it. The proposed method to accomplish this is to have some sort of DEM agent on each work station involved in an exercise which can contact the DEM monitor when the machine is being used. This will prevent a work station contained in the DEM config list being reported as down when it is really just not being used at the time.

With the amount of work stations being monitored for various performance metrics during the STOW FSTs and the ACTD, the DEM operators encountered information overload. It is recommended that DEM investigate alternate methods of presenting monitored information giving the operator more flexibility to “customize” the displayed information. DEM will evaluate available cots S/W packages and look into developing the current DEM GUI into a more flexible design with hierarchical alarm reporting in mind.

The Informix data base used by DEM was very good at collecting the monitored data for DEM during a simulation exercise. It was primarily designed for real time data collection and After Action Review (AAR) data queries. It is quite capable of this task, however, during STOW FST testing and the ACTD there were times when real time data queries were asked for. When running a query on the data base at the same time it was collecting data, the queries took a long time (up to 15-20 minutes). This was too long for real time data analysis. For this reason, DEM will investigate other methods for presenting real time data to the operator. As a matter of fact, because of cost factors, DEM will re-look the data base scheme it is currently using altogether. One possibility is to log data into flat files and allow the end user to chose which ever data base they would like to for AAR. More efficient ways to display real time data also need to be investigated.

Author's Biographies

Bernard Gajkowski is a Project Director within the Engineering Synthetic Environment Office of STRICOM, Orlando, FL. Currently he is a Project Director of several research and development efforts such as the DiSECT project, the Federation Test System (FTS), and Course of Action Analysis.

Greg Schow is the Systems Engineer within the Engineering Distributed Interactive Simulation Division of STRICOM, Orlando, FL. He was the lead systems engineer for the DiSECT program during the support of the DARPA STOW program.

John Pfalz is an integration engineer at TASC in San Antonio, TX. He is the Deputy Delivery Order Manager of the DEM and AAR portions of the DiSECT program. John has been involved in Simnet and DIS based programs, as well as the HLA based program he is currently working on.

Jeff Swauger is a Senior Engineer at TASC in Orlando, FL. He is the technical director for TASC's M&S SEI contract with STRICOM and was the previous delivery order manager of the DiSECT program.

Harvey Meier is a Senior Software Engineer for TASC in Orlando, FL. He is the Delivery Order Manager for the Distributed Simulation Exercise Construction Toolset (DiSECT) program.

Chris Gullette is a Software Engineer for TASC in Orlando, FL. He is the development team leader for the Distributed Exercise Management (DEM) section of the DiSECT project.